LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-409943

Virtual Array Receiver Options for 64-ary Pulse Position Modulation (PPM)

A. J. Mendez, V. J. Hernandez, R. M. Gagliardi, C. V. Bennett

January 20, 2009

SPIE Photonics West
San Jose, CA, United States
January 24, 2009 through January 29, 2009

Disclaimer

Virtual Array Receiver Options for 64-ary Pulse Position Modulation (PPM)

Antonio J. Mendez[a], Vincent J. Hernandez[b], Robert M. Gagliardi[c], and Corey V. Bennett[b]
[a]Mendez R&D Associates, P.O. Box 2756, El Segundo, CA 90245;
[b]Lawrence Livermore National Laboratory, P.O. Box 808, L-223, Livermore, CA 94551;
[c]Dept. of Electrical Engineering, University of Southern California, Los Angeles, CA 90089

ABSTRACT

NASA is developing technology for 64-ary PPM using relatively large PPM time slots (10 ns) and relatively simple electronic-based receiver logic. In this paper we describe photonics-based receiver options for the case of much higher data rates and inherently shorter decision times. The receivers take the form of virtual (array or quadrant) arrays with associated comparison tests. Previously we explored this concept for 4-ary and 16-ary PPM at data rates of up to 10 Gb/s. The lessons learned are applied to the case of 64-ary PPM at 1.25 Gb/s. Various receiver designs are compared, and the optimum design, based on virtual arrays, is evaluated using numerical simulations.

Keywords: Pulse position modulation; optical communications; arrayed receivers

1. INTRODUCTION

PPM signaling has the advantages of *M*-ary signaling with power efficiency and use of direct detection receivers[1-3]. Currently, NASA is developing technology to implement 64-ary PPM[4], but it uses relatively large time slots (10 ns) so that the PPM slot decision can be processed by relatively simple (and low bandwidth) electronic receiver logic. In this paper we explore receiver options for the case of higher transmission rates (> 1Gb/s), which are accompanied by inherently shorter PPM slot times, require faster decision times, and conceivably place greater demands on the receiver's electronic bandwidth. For these cases, we have been exploring decision aids in the form of a virtual array receiver, where optical processing incorporated into the slot comparison tests alleviates the bandwidth demands on the electronics. In a virtual array receiver, *M* optical copies of a received PPM frame are incrementally delayed and uniquely combined so that all slots within the frame are presented simultaneously to the final electronic decision logic, a form of time-to-space mapping. The net result allows the decision logic to operate at the frame rate, rather than the slot rate. Designs for simpler 4-ary[5] and 16-ary[6] PPM virtual receivers have been captured with numerical simulations for input data rates of up to 10 Gb/s. These have yielded symbol error rate calculations that take the form of constellation plots. A hardware demonstration of 4-ary PPM at 1.25 Gb/s has also been demonstrated[7]. Now, the lessons learned from these examples are applied to the case of 64-ary PPM at > 1 Gb/s. To begin, we first present a suitable implementation for a PPM encoder that compliments the receivers that will be discussed. Afterwards, the paper compares various receiver options, especially virtual quadrant receivers and virtual array receivers, focusing on their different methods of performing time-to-space mapping and their associated control laws and comparison tests. A feasible implementation of a virtual array receiver is proposed and captured using numerical simulations.

2. PPM ENCODING

Before discussing receiver designs for >1 Gb/s 64-ary PPM, it is appropriate to first consider techniques for encoding (transmitting) PPM signals. Similar to the receivers, PPM encoders at higher data rates likewise face electronic bandwidth limitations as the data rate increases. It is therefore beneficial to incorporate optical components into their design to relieve the bandwidth requirements. One such design is proposed here. In general, the PPM pulse encoder maps an inputted data sequence to one of *M* contiguous, non-overlapping pulse positions that constitute a data frame, where the frames themselves are contiguous but may include a guard time for synchronization purposes. The data sequence consists of $N=log_2(M)$ bits and its numerical value may be expressed as

$$K = \sum_{i=0}^{N-1} a_i \cdot 2^i \tag{1}$$

where a_i is the transmitted binary symbol (0,1) in the sequence. If K is made to correspond to the number of unit pulse shifts (0 to M-1) applied to the incoming pulse, then the encoder may be implemented using a sequence of serial switches and parallel delay lines, as illustrated in Figure 1 for M=64. The figure depicts discrete components and assumes an external mode-locked laser source, but eventually a planar lightwave circuit (PLC) implementation should be incorporated to reduce its dimensions, minimize insertion losses, and achieve the precision required for the various stages. In Figure 1, each switch is a dual output Mach-Zehnder Interferometer (MZI) modulator controlled by the data bit a_i. The value of a_i determines whether the pulse is sent through a delay line containing a relative delay of either 0 or 2^i. For M=64 at 1.25 Gb/s (1 Gb/s with 8B/10B line coding), six such modulator/delay line pairs are required, with PPM frame times of 4800 ps and slot times of 75 ps. Each modulator is driven by the bits of the demultiplexed input bit sequence at a rate of 208.33 MHz, and each parallel delay line can apply a delay of 2^i x 75 ps. The required input pulse train for the encoder must have pulse widths of <75 ps to fit within the slots and a repetition rate of 208.33 MHz to correspond to the frame size.

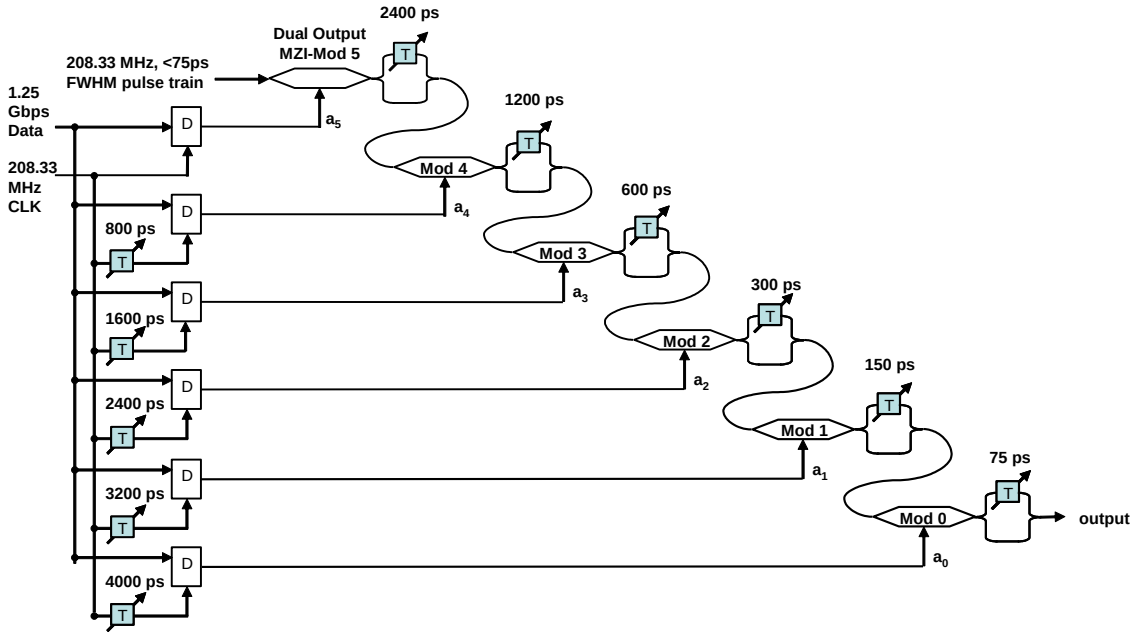


Figure 1. Architecture of the 64-ary PPM Encoder at 1.25 Gb/s utilizing cascaded dual output Mach-Zehnder interferometer modulators (MZI-Mod).

Using Eqn. (1), the data sequence 000010 would be mapped to slot value K=*2*, as shown in Figure 2. The fact that the bit sequence to pulse position is unique, as translated by the switch positions and associated delays, is demonstrated by Table I, where all of the possible 64 bit sequences for N=6 are shown to correspond to unique switch states and slot values.

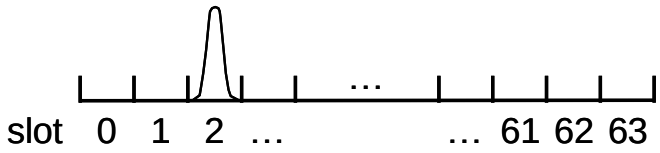


Figure 2. Example of Equation 2: Pulse Position for the Data Sequence 000010.

Table 1. Unique Correspondence of Bit Sequence, Modulator (Mod) Switch State, and Slot Value.

Transmitted bit sequence						Slot position after Mod 5	Slot position after Mod 4	Slot position after Mod 3	Slot position after Mod 2	Slot position after Mod 1	Slot position after Mod 0
a_5	a_4	a_3	a_2	a_1	a_0						
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	1	0	0	0	0	0	2	2
0	0	0	0	1	1	0	0	0	0	2	3
0	0	0	1	0	0	0	0	0	4	4	4
0	0	0	1	0	1	0	0	0	4	4	5
0	0	0	1	1	0	0	0	0	4	6	6
0	0	0	1	1	1	0	0	0	4	6	7
0	0	1	0	0	0	0	0	8	8	8	8
0	0	1	0	0	1	0	0	8	8	8	9
0	0	1	0	1	0	0	0	8	8	10	10
0	0	1	0	1	1	0	0	8	8	10	11
0	0	1	1	0	0	0	0	8	12	12	12
0	0	1	1	0	1	0	0	8	12	12	13
0	0	1	1	1	0	0	0	8	12	14	14
0	0	1	1	1	1	0	0	8	12	14	15
0	1	0	0	0	0	0	16	16	16	16	16
0	1	0	0	0	1	0	16	16	16	16	17
0	1	0	0	1	0	0	16	16	16	18	18
0	1	0	0	1	1	0	16	16	16	18	19
0	1	0	1	0	0	0	16	16	20	20	20
0	1	0	1	0	1	0	16	16	20	20	21
0	1	0	1	1	0	0	16	16	20	22	22
0	1	0	1	1	1	0	16	16	20	22	23
0	1	1	0	0	0	0	16	24	24	24	24
0	1	1	0	0	1	0	16	24	24	24	25
0	1	1	0	1	0	0	16	24	24	26	26
0	1	1	0	1	1	0	16	24	24	26	27
0	1	1	1	0	0	0	16	24	28	28	28
0	1	1	1	0	1	0	16	24	28	28	29
0	1	1	1	1	0	0	16	24	28	30	30
0	1	1	1	1	1	0	16	24	28	30	31
1	0	0	0	0	0	32	32	32	32	32	32
1	0	0	0	0	1	32	32	32	32	32	33
1	0	0	0	1	0	32	32	32	32	34	34
1	0	0	0	1	1	32	32	32	32	34	35
1	0	0	1	0	0	32	32	32	36	36	36
1	0	0	1	0	1	32	32	32	36	36	37
1	0	0	1	1	0	32	32	32	36	38	38
1	0	0	1	1	1	32	32	32	36	38	39
1	0	1	0	0	0	32	32	40	40	40	40
1	0	1	0	0	1	32	32	40	40	40	41
1	0	1	0	1	0	32	32	40	40	42	42
1	0	1	0	1	1	32	32	40	40	42	43
1	0	1	1	0	0	32	32	40	44	44	44
1	0	1	1	0	1	32	32	40	44	44	45
1	0	1	1	1	0	32	32	40	44	46	46
1	0	1	1	1	1	32	32	40	44	46	47
1	1	0	0	0	0	32	48	48	48	48	48
1	1	0	0	0	1	32	48	48	48	48	49
1	1	0	0	1	0	32	48	48	48	50	50
1	1	0	0	1	1	32	48	48	48	50	51
1	1	0	1	0	0	32	48	48	52	52	52
1	1	0	1	0	1	32	48	48	52	52	53
1	1	0	1	1	0	32	48	48	52	54	54
1	1	0	1	1	1	32	48	48	52	54	55
1	1	1	0	0	0	32	48	56	56	56	56
1	1	1	0	0	1	32	48	56	56	56	57
1	1	1	0	1	0	32	48	56	56	58	58
1	1	1	0	1	1	32	48	56	56	58	59
1	1	1	1	0	0	32	48	56	60	60	60
1	1	1	1	0	1	32	48	56	60	60	61
1	1	1	1	1	0	32	48	56	60	62	62
1	1	1	1	1	1	32	48	56	60	62	63

3. PPM RECEIVER OPTIONS

The previous section showed that there is a unique way of defining the PPM pulse position corresponding to an input data sequence for any *M*. This leads to a straightforward way of defining the architecture for the PPM encoder. Unfortunately, the architecture for the PPM decoder/receiver is not as obvious, especially as *M* gets large and/or the input data rate gets high (> 1 Gb/s). In this section we'll describe a few options for PPM receiver architectures, and then focus on a virtual array receiver, the optimum design from the viewpoint of minimal electronic bandwidth requirement, reduced complexity (especially photodetector count), and growth potential to higher *M* and input data rates.

The point of departure is the traditional PPM receiver that uses a single detector to sequentially sample all *M* slots[1]. It is followed by logic-based decision circuitry to determine which slot contains the pulse, and that slot estimate converts to an output data sequence by means of look-up tables. As previously stated, this technique requires the detector and subsequent electronics to operate at the slot rate, and thus this option becomes increasingly difficult to implement at higher *M* and data rates due to electronic bandwidth limitations. At the other extreme, it is conceivable to have a PPM receiver that employs an *M* detector array, where each slot of the frame is time-to-space or time-to-space-to-frequency[8] mapped into its own detector, as shown in Figure 3a. Subsequent logic and look-up tables then determine the most likely occupied slot and converts it into a data sequence. This technique uses an initial 1:*M* optical splitter that provides the incoming frame to each detector. The detectors sample incoming frames simultaneously, and thus each frame is incrementally delayed so that each detector samples a different slot of the frame, designated [*s0, …, s(M-1)*], during the sample time. In this way, only the detector and sampler of the receiver need to operate at the slot rate. All subsequent components, including the decision logic, need only operate at the frame rate. Furthermore, if an optical gate[9] (not shown) can be incorporated before the detectors, then the detector and all subsequent electronics can operate at the frame rate. The reduced bandwidth demands comes at the expense of requiring a large number of discrete detectors. In fact, as many as 2*M* detectors has been suggested for a proposed time-to-frequency-to-space mapping scheme for 100 Gb/s PPM.[8] The next receivers discussed seek to minimize the detector count. Analogous to laser spot tracker concepts, the virtual quadrant and array receivers employ optical processing that perform decision schemes or control laws, mapping the slots into a virtual array to estimate the occupied slot.

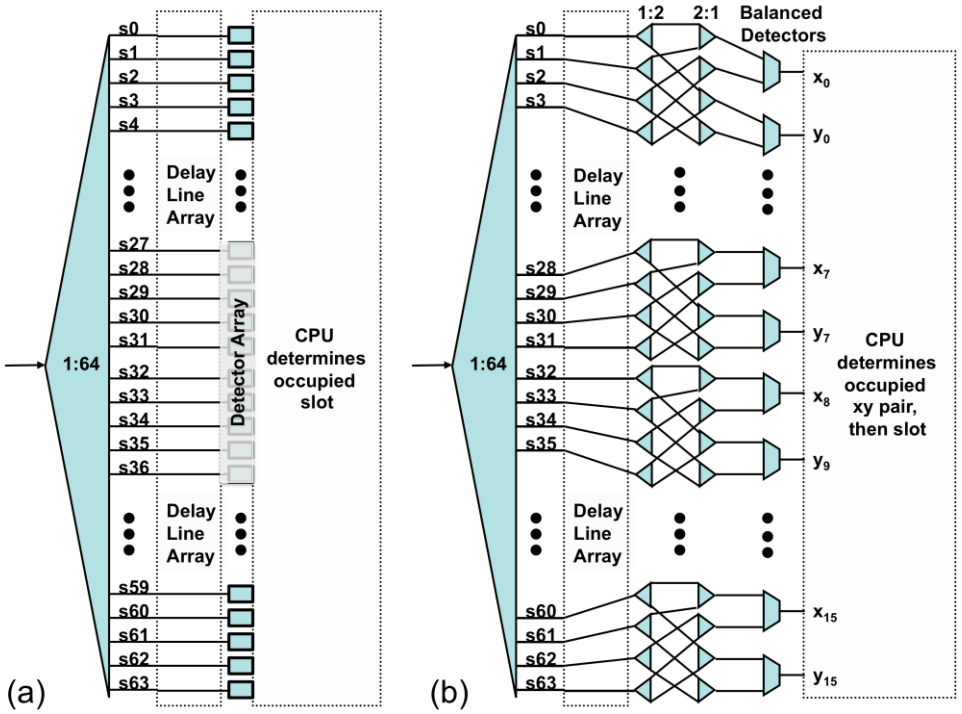


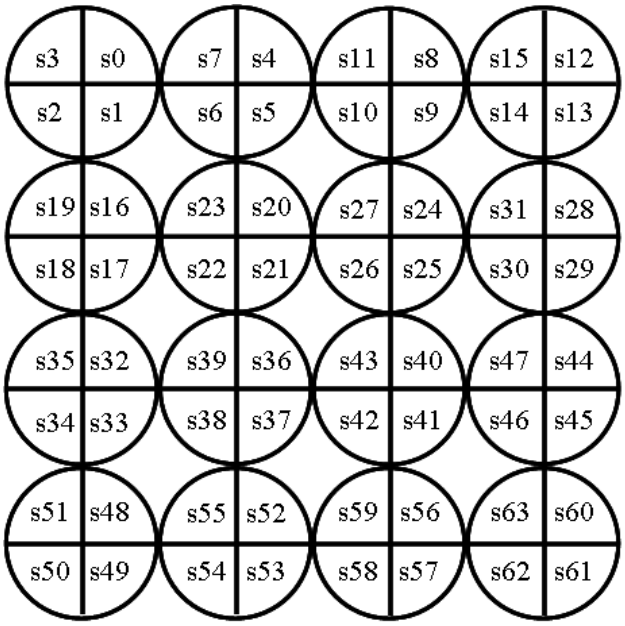
Figure 3. Architecture for (a) *M* arrayed detector and (b) virtual quadrant receivers.

The virtual quadrant receiver (Figure 3b) is an extension of time-to-space mapping that has been demonstrated in hardware for 4-ary PPM at 1.25 Gb/s for optical fiber communications[7]. Similar to the *M*-arrayed detector architecture, optical splitters and delay lines are used to copy and delay the incoming frame, with the intention of having a detector array that samples all slots simultaneously. The key difference is that a comparison test, or control law, implemented in the optical domain is now employed to aid the slot decision. The law used in the 4-ary case is given as

$$[x_0 = (s0 + s1) - (s2 + s3),\ y_0 = (s0 + s3) - (s1 + s2)], \quad (2)$$

where *sK* denotes the slot that will be sampled by the detector. As can be followed after the first four splitter outputs in Figure 3b (marked *s0* to *s3*), optical combiners perform the control law addition while two differential detectors perform subtraction. The differential detectors outputs, x_0 and y_0, can then be plotted onto a coordinate system where each quadrant represents a slot position. The 64-ary PPM implementation merely applies the same control law on subsets of the slots, with the resulting time-to-space mapping shown in Figure 4a. Each encircled 2x2 cell array represents the basic mapping for 4-ary PPM applied to all 64 slots (four slots for each cell). Subsequent electronics following the differential receiver pairs could then determine which cell and slot contained the pulse. The overall advantage gained from using the control law is the minimization of required detectors. The virtual quadrant receiver requires a total of 32 differential detectors, two for each of the sixteen cells, a twofold improvement over the *M* detector arrayed receiver.

The virtual array receiver further minimizes the detector count. These receivers map the slots into a $\sqrt{M}$x$\sqrt{M}$ array as illustrated in Figure 4b. Then, one need only to find the coordinates of the cell that has the highest energy in it to determine the occupied slot. The control laws for determining the coordinates are summarized in Table II, where rows (and columns) are grouped into pairs. Slot addition and subtraction are again performed respectively using optical combiners and differential receivers. A total of $\sqrt{M}$ laws produce normalized output given as 1, -1, or 0 (if neither row is occupied). This procedure gives the row and column coordinates of the cell containing the highest energy and can be used in conjunction with look-up tables to identify the transmitted bit sequence. The total number of detectors for the virtual array receiver is only $\sqrt{M}$=8, substantially less required than the *M* detector array or virtual quadrant receivers; it therefore represents the optimal solution.



(a)

s0	s1	s2	s3	s4	s5	s6	s7
s8	s9	s10	s11	s12	s13	s14	s15
s16	s17	s18	s19	s20	s21	s22	s23
s24	s25	s26	s27	s28	s29	s30	s31
s32	s33	s34	s35	s36	s37	s38	s39
s40	s41	s42	s43	s44	s45	s46	s47
s48	s49	s50	s51	s52	s53	s54	s55
s56	s57	s58	s59	s60	s61	s62	s63

(b)

Figure 4. Time-to-space mapping of 64-ary PPM slots using (a) virtual quadrant receivers and (b) virtual array receivers.

Table II. Virtual Array Receiver Control Laws for Identifying the PPM Slot Row/Column Coordinates.

Control Law	Result = 1	Result = -1	Result = 0
(s0+s1+s2+s3+s4+s5+s6+s7)-(s8+s9+s10+s11+s12+s13+s14+s15)	Row 1	Row 2	
(s16+s17+s18+s19+s20+s21+s22+s23)-(s24+s25+s26+s27+s28+s29+s30+s31)	Row 3	Row 4	
(s32+s33+s34+s35+s36+s37+s38+s39)-(s40+s41+s42+s43+s44+s45+s46+s47)	Row 5	Row 6	
(s48+s49+s50+s51+s52+s53+s54+s55)-(s56+s57+s58+s59+s60+s61+s62+s63)	Row 7	Row 8	Pulse is contained in another row/column pair.
(s0+s8+s16+s24+s32+s40+s48+s56)-(s1+s9+s17+s25+s33+s41+s49+s57)	Column 1	Column 2	
(s2+s10+s18+s26+s34+s42+s50+s58)-(s3+s11+s19+s27+s35+s43+s51+s59)	Column 3	Column 4	
(s4+s12+s20+s28+s36+s44+s52+s60)-(s5+s13+s21+s29+s37+s45+s53+s61)	Column 5	Column 6	
(s6+s14+s22+s30+s38+s46+s54+s62)-(s7+s15+s23+s31+s39+s47+s55+s63)	Column 7	Column 8	

4. VIRTUAL ARRAY RECEIVER ARCHITECTURE AND NUMERICAL SIMULATIONS

This section further evaluates the virtual array receiver design using numerical simulation. A feasible virtual array receiver architecture is shown in Figure 5 utilizing planar lightwave circuits (PLCs). The splitting stage cascades a 1:2 splitter and two 1:32 splitters, creating 64 copies of the input. Each copy is designated to represent a slot of the frame, and a subsequent delay line array synchronizes the images so that all of the slots are temporally aligned. The slot-synchronized images are further split in order to provide inputs to the row and column processing defined in Table II. Note that erbium doped waveguide amplifiers (EDWAs)[10] are placed between the splitting cascades. These are critical for minimizing insertion losses and enable the large splitting ratio. The design does not yet incorporate optical gates, as 64-ary PPM at 1.25 G5 Gb/s corresponds to a 13.3 GHz slot rate, falling within the specification of many detectors.

The architecture in turn is captured using RSoft's OptSIM, as driven by the PPM transmitter in Figure 1 under a back-to-back configuration. The EDWAs are modeled after those used in 4-ary PPM virtual array receiver demonstrations, with a small signal gain of 20 dB, saturation power at 10 dBm, and noise figure of 7 dB. The total insertion loss from the input to a differential detector is 12 dB, considering splitting and combining losses and EDWA gain. The differential detectors are modeled off of u^2t BPRV2123 43 Gb/s DPSK balanced photoreceivers which have a 22 GHz 3dB cutoff bandwidth. A total of 15240 bits are encoded into 2540 PPM symbols and sent into the receiver at -10 dBm input power. The photodetector sampled outputs produced by the simulation are then processed with Matlab to map the slot decision into an array that resembles a constellation plot, as shown in Figure 6a. Variances from the center of each array cell arise from noise contributed by the EDWA amplified spontaneous emission, detector thermal noise, and detector shot noise. None of the sampled output points cross into another cell, indicating that no errors occur for the transmitted symbols. Lowering the input power to the receiver to -11 dBm yields the result in Figure 6b, where 32 symbol errors occur, as indicated by the sample points that cross into other cells. It is interesting to note that errors do not gradually exceed the cell boundaries, but instead completely leap into another cell when an incorrect decision is made. The overall result indicates that the sensitivity of the virtual array receiver is near -10 dBm for this particular architecture. The result can be expected to improve with incorporation of forward error correction encoding and decoding, respectively performed before the PPM encoder and after the PPM decoder.

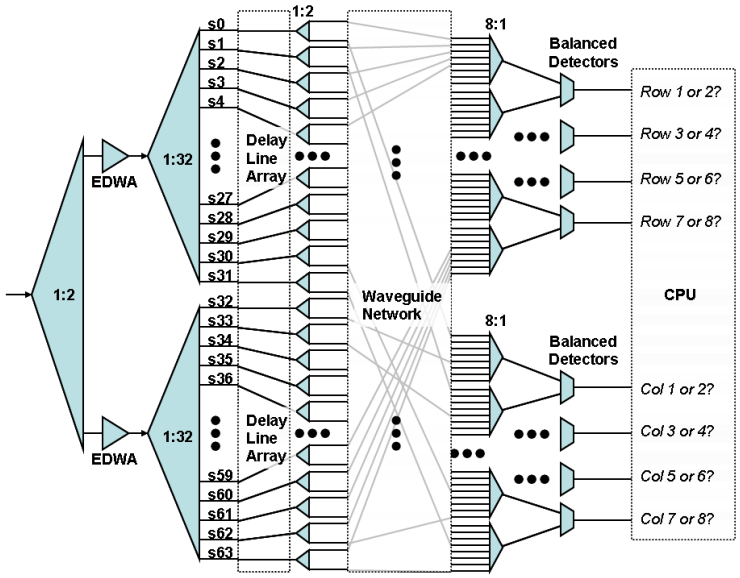


Figure 5. Means of Implementing the Virtual Array Receiver by Means of Planar Lightwave Circuits (PLCs).

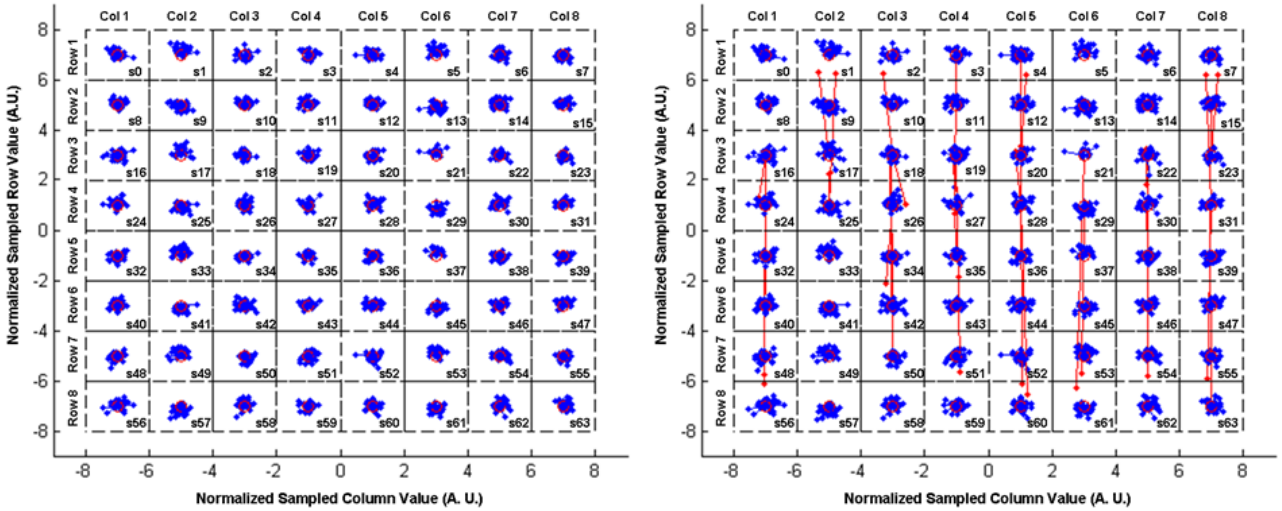


Figure 6. Numerical simulation results for the virtual array receiver for 2540 PPM symbols (15240 bits). No symbol errors occur for -10 dBm input into the receiver (left) while thirty-two errors occur at -11 dBm (right).

5. REMARKS AND CONCLUSIONS

We have described methods for transmitting (encoding) and receiving 64-ary PPM at 1.25 Gb/s An algorithm for transforming any transmitted bit sequence to an equivalent pulse position in a PPM frame has been given, and this algorithm translates to an architecture consisting of a unique set of dual output switches and delays for any bit sequence length and transmitted bit rate. A corresponding PPM receiver architecture is less straightforward to determine, and several options have been discussed. The optimal design, the virtual array receiver, is based on $\sqrt{M}$ differential receivers for the case of M-ary PPM. With incorporation of optical gating, the outputs can be sampled at the frame rate, alleviating the bandwidth requirement on the electronics. Its associated control laws give estimates of the best-fit row and column coordinates corresponding to the slot position. Associated look up tables then interpret the input bit sequence. The actual receiver implementation can be based on PLCs incorporating EDWA technology, and a sample implementation has been numerically simulated. It can be shown that error-free symbol and error rates can be achieved for the given sample inputs. Future work involves combining all of these concepts and architectures into compact, integrated subassemblies.

ACKNOWLEDGEMENT

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-7NA27344. The work was funded in part by DARPA SBIR Phase II Adoption contract W31P4Q-05-C-R161. Release number LLNL-CONF-409943.

REFERENCES

[1] Gagliardi, R. M. and Karp, S., [Optical Communications], 2nd ed., John Wiley and Sons, New York, (1995).

[2] Armstrong, J., "OFDM: from copper and wireless to optical," Proc. IEEE Optical Fiber Communications Conf. (OFC), 1499-525, slides 41-44 (2008).

[3] Narimanov, E., in [Optical Code Division Multiple Access: fundamentals and applications], Prucnal, P.R., Ed., CRC Taylor & Francis, Boca Raton, FL., 106 (2006).

[4] Krainak, M. A., Chen, J. R., Dabney, P. W., Ferrara, J. F., Fong, W. H., Martino, A. J., McGarry, J. F., Merkowitz, S. M., Principe, C. M., Sun, X. and Zagwodzki, T. W., "Direct-detection free-space laser transceiver test-bed," Proc. SPIE 6877, 687703-1-12 (2008).

[5] Mendez, A. J., Hernandez, V. J., Gagliardi, R. M. and Bennett, C. V., "Design and Evaluation of a Virtual Quadrant Receiver for 4-ary Pulse Position Modulation/Optical Code Division Multiple Access (4-ary PPM/O-CDMA)," Proc. SPIE 6457, 64570H-1-6 (2007).

[6] Mendez, A. J., Gagliardi, R. M., Hernandez, V. J. and Bennett, C. V., "Receiver Architecture for 12.5 Gb/s 16-ary Pulse Position Modulation (PPM) Signaling," Proc. IEEE Avionics, Fiber-Optics, and Photonics Technology Conf. (AVFOP), Paper WB4 (2008).

[7] Hernandez, V. J., Mendez, A. J., Gagliardi, R. M., Bennett, C. V. and Lennon, W. J., "Performance Impact of Multiple Access Interference in a 4-ary Pulse Postion Modulated Optical Code Division Multiple Access (PPM/O-CDMA) System," Proc. IEEE Optical Fiber Communications Conf. (OFC), 1652-4 (2008).

[8] Birnbaum, K. M. and Farr, W. H., "Pulse position modulation/demodulation with picosecond slot widths," Proc. SPIE 6877, pp. 68770K-8 (2008).

[9] Chan, V. W. S., Hall, K. L., Modiano, E. and Rauschenbach, K. A., "Architectures and technologies for high-speed optical data networks," J. Lightw. Technol. 16(12), 2146-68 (1998).

[10] Frolov, S., Tek-Ming, S. and Bruce, A. J., "EDWA: key enabler of optical integration on PLC," Proc. SPIE 4990, 47-54 (2003).